1103179-0009

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant	: Edward Colles Nevill
Serial No.	: 10/066,475
Filed	: February 1, 2002
For	: INTEROPERABILITY WITH MULTIPLE INSTRUCTION SETS
Examiner	: Kenneth R. Coulter
Group Art Unit	: 2141

CERTIFICATE OF TRANSMISSION UNDER 37 C.F.R. 1.8

I hereby certify that this paper is being facsimile transmitted to the U.S. Patent and Trademark Office on the date indicated below at the facsimile number 571-273-8300.

John Scheibeler — Name of Sender
35,346 — Reg. No.
Signature
July 11, 2008 — Date of Signature

Mail Stop Appeal Brief - Patents
Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

FACSIMILE NO: **571-273-8300**
DATE: **July 11, 2008**
PAGES: **16 pages (excluding accompanying papers)**

REPLY BRIEF

Sir:

Applicant hereby submits this Reply Brief in response to an Examiner's Answer, dated May 14, 2008. Applicant also submits the accompanying Request for Oral Hearing in duplicate copy. The Commissioner is hereby authorized to charge the $1030.00 fee for the Request for Oral Hearing and any other fees associated with the filing of this Reply Brief to Deposit Account No. 23-1703.

07/14/2008 VBUI11 00000010 231703 10066475
01 FC:1403 1030.00 DA

(1) STATUS OF CLAIMS

Claims 1-70 are pending in the reissue application.

Claims 1-14 were originally allowed in Application Serial No. 08/840,557, now U.S. Patent No. 6,021,265, of which the present application is a reissue. Claims 15-64 were added by a Preliminary Amendment filed with the reissue application on February 1, 2002. Claims 65-70 were added by an Amendment originally filed on December 17, 2004, and filed again, on June 3, 2005, in response to a Notice of Non-Compliant Amendment.

This is an appeal from a second, non-final Office Action dated February 15, 2006 ("the 2/15/06 Office Action"), which rejected all pending claims of the above-referenced application. Claims 1-70 stand rejected under 35 U.S.C. § 102(b) as anticipated by U.S. Patent No. 5,115,500 to Larsen.

Applicant filed a Notice of Appeal and a Pre-Appeal Brief Request for Review on June 15, 2006, with the corresponding fee. Applicant received a Notice of Panel Decision from Pre-Appeal Brief Review, mailed October 25, 2006, indicating that Applicant should proceed to the Board of Patent Appeals and Interferences ("the Board") with its appeal of the rejection of claims 1-70. Applicant filed an Appeal Brief on November 27, 2006. The Examiner responded in an Examiner's Answer, dated May 14, 2008.

(2) GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

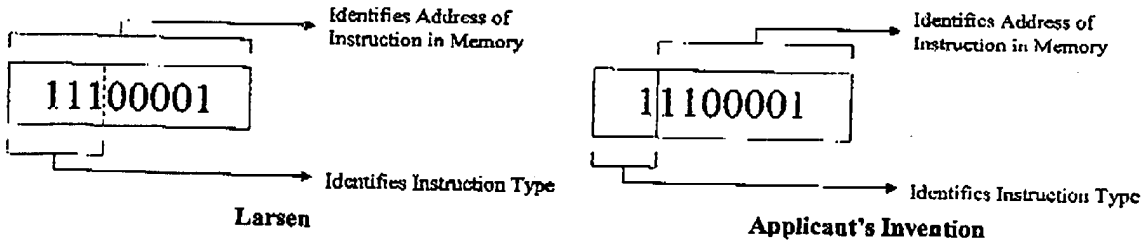
Whether claims 1-70 are unpatentable under 35 U.S.C. § 102(b) as allegedly anticipated by U.S. Patent No. 5,115,500 to Larsen.

(3) ARGUMENT

I. Introduction

Although there are numerous differences among the pending claims, this appeal can be decided with respect to all claims based on a single issue. In U.S. Patent No. 5,115,500 to Larsen ("Larsen"), due to the advance partitioning of memory by instruction type, at least a portion of the sequence of bits used to identify the memory address of an instruction **must** be re-used to identify the instruction type, otherwise the system of Larsen will fail to operate correctly. On the other hand, each and every claim of the present application requires, albeit in different implementations, that the bits that specify the address of an instruction are **independent** of the bit or bits that specify the instruction type (i.e., no address bits are re-used to specify the instruction type in Applicant's invention).

The diagrams below compare how Larsen uses a sequence of bits stored in a register with how that same sequence of bits would be used according to Applicant's claimed invention[1]:



In Larsen, the first three bits are **both** (1) used to identify an instruction type **and** (2) included as part of the overall memory address for the instruction. In the above example of Applicant's invention, the first bit is used only to identify an instruction type. Unlike Larsen, that bit is not

[1] In this example, Applicant's instruction type identifying bit is located in the highest order position. As described in the specification, this bit may be located in several places (the exact position of the bit is not critical to the invention), but in each case the bit that is used to identify the instruction type is not used as part of the address of the instruction.

also used as part of the address of the instruction. Applicant's claimed invention thus avoids the need to partition memory in advance.

Not only does Larsen not disclose independence between the bit(s) that identify instruction type and the bits that identify an instruction address, but such independence is not even possible in the Larsen system. If Larsen were to operate as the Examiner suggests it would fail to operate correctly.

II. If Operated According To Applicant's Claimed Invention, Larsen Would Fail To Operate Correctly

A. Larsen Re-uses Certain Address Bits To Specify The Instruction Type

Larsen requires the partitioning of memory in advance such that an instruction's type is defined by its location in memory, i.e., its address. (Larsen, col. 6, li. 55-66.) For example, instructions of a first type are placed in certain partitions reserved for that first type while instructions of a second type are placed in certain other partitions reserved for that second type. To properly decode an instruction, the decoding circuitry must therefore know at least a portion of the instruction's memory address, i.e., the memory partition from which it was fetched. In other words, the address of an instruction in Larsen necessarily must include bits that also identify the instruction's type.

This is shown in Figure 2 of Larsen, reproduced, in relevant part, below:

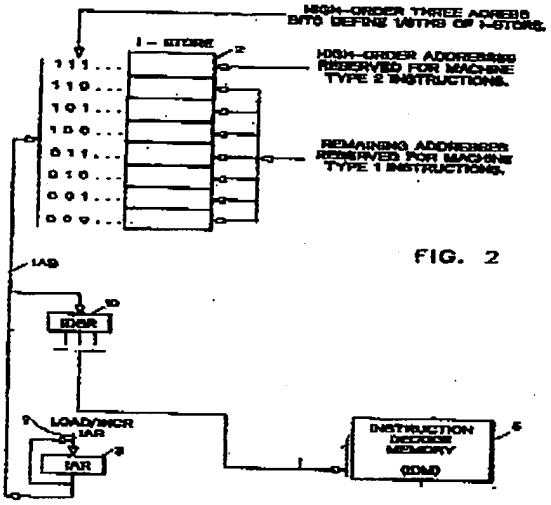


FIG. 2

According to Larsen, the instruction address register (IAR **3**), contains the address of the next instruction to be fetched from memory (the memory is referred to as I-store **2**). (Larsen, col. 3, li. 31-33, 59-62.) Figure 2 shows that the bits from IAR **3** are output **both** to I-store **2** (where all the bits from IAR **3** are used to identify the memory address of an instruction) and to IDSR **10** (where at least the three highest-order bits are re-used to identify the instruction's type). These re-used high-order bits of the address are sent from IDSR **10** to IDM **5** where they are used to specify the instruction type so the fetched instruction can be properly decoded. As Larsen states, "[d]ecoding of any specific instruction [fetched from I-store **2**] thus depends not only on the [instruction itself] but on the region of the I-store **2** from which the instruction was fetched. This **portion of the address** is provided by the IDSR **10** output." (Larsen, col. 6, li. 11-16 (emphasis added).)

In the example shown in Figure 2 of Larsen, the memory, i.e., I-store **2**, is partitioned into eight equal-length sections. The highest-order partition contains type 2 instructions while the seven lowest-order partitions contain type 1 instructions. (Larsen, col. 6, li. 55-60.) Therefore,

to properly decode an instruction, Larsen's instruction decoding circuitry (IDM **5**) needs to know the partition from which the instruction was fetched. This information is specified by the three highest-order bits of the instruction's address (stored in IDSR **10**). (Larsen, col. 5, li. 54 – col. 6, li. 39.)

Thus, Larsen must re-use at least some of the same bits that are used to specify the address of a given instruction in memory (I-store **2**) to specify the instruction type. As shown in the next section, if, as the Examiner appears to assume, these bits are not re-used, the system described by Larsen would fail.

B. Larsen Would Fail If At Least Some Instruction Address Bits Were Not Re-used To Specify The Instruction Type

In this section, Applicant demonstrates how the apparatus described in Larsen would fail to operate properly if it did not re-use bits from an instruction's address to specify the instruction set or instruction type for decoding (as suggested by the Examiner's rejection). For present purposes, Applicant will use an example from Larsen's Figure 2 where the instruction address points to a type 2 instruction, defined by the three highest-order address bits being 111 (Larsen represents the remaining address bits as ellipses or . . .). For purposes of illustration, if Applicant assumes that the bits represented by the ellipses are 00001, the complete address of the instruction is 11100001. This value addresses a single specific memory location in the portion of I-store **2** reserved for type 2 instructions.

The Examiner asserts that "the bits that are not the top 3 bits from the IAR" in Larsen, 00001 in the above example, disclose a subset of bits that identifies an address of an instruction. (Examiner's Answer, p. 15.) The Examiner similarly states that "[t]he address of the next instruction can be considered the bits that are not the top 3 bits from the IAR" of Larsen (again, 00001 in the above example). (Examiner's Answer, p. 17.) The assertion that the bits that are

"not the top 3 bits" (the remaining five bits or 00001) identify an address of an instruction is wrong because these bits only provide part of an address and are ambiguous at best. To complete the address, three additional bits are needed (111 in this example). If "the top 3 bits" of IAR 3 are ignored, as the Examiner suggests, these three bits could be any combination of ones and zeros.

There are eight possible combinations for the three highest-order bits (000, 001, 010, 011, 100, 101, 110, 111), but only one of these combinations would represent the correct address of the next instruction (11100001). There is no way to predict which of these eight possible combinations might occur in Larsen if the top three bits were ignored. Thus, if the system in Larsen operated in the manner suggested by the Examiner, the Larsen system could access the wrong memory location in I-store 2 at least seven out of every eight occurrences, or 87.5% of the time. Even this error rate is uncertain because, if the three highest-order bits of IAR 3 are not supplied to the addressing circuitry in Larsen, the addressing circuitry may retain values from prior cycles. If the retained values were anything other than 111, the system in Larsen would retrieve the wrong instruction over and over again. Such a system could not possibly function.

Simply put, in order to operate correctly, Larsen **requires** that the entire sequence of bits in IAR 3 be used to identify the address for an instruction stored in I-store 2. This necessarily includes the three highest-order bits that are used to specify the instruction type. If, as required by the claims of the present application, Larsen's instruction addressing circuitry attempted to not use, disregard or be non-responsive to the three highest-order bits from IAR 3, which identify the instruction type, the system of Larsen would identify an incorrect address, fetch the wrong instruction, and decode the fetched instruction improperly.

Applicant's Appeal Brief explains that this difference between Larsen and the present application is not trivial. (Appeal Brief, p. 19.) The present invention, as set forth in the pending claims, provides flexibility to a software designer who can use memory more efficiently by storing instructions as he or she sees fit, according to the needs of the software design. By requiring independence between the bits used to specify an instruction address and the bits used to specify an instruction type, the claims of the present invention avoid the rigid partitioning of memory based on instruction type as required by Larsen. Accordingly, not only does Larsen not disclose each and every limitation of Applicant's pending claims, if the system of Larsen was to operate in the manner suggested by the Examiner, it would be a major failure.

III. All Pending Claims Contain Limitations Reflecting That The Bits That Specify The Address Of An Instruction And The Bits That Specify Instruction Type Are Independent Of One Another

As set forth in the preceding section, Larsen does not disclose a sequence of bits having a portion that specifies the address of an instruction and an **independent** portion that specifies an instruction type. The Examiner asserts that the actual language of Applicant's claims is broad enough to cover the system of Larsen, in which the three highest-order bits from IAR 3 both specify at least a portion of the address of an instruction and specify the instruction type. (Examiner's Answer, pp. 16-17.)

Applicant respectfully submits that the Examiner is wrong. Although there are other differences between Applicant's invention and Larsen, as set forth more fully below, each claim in the pending Application contains a limitation that Larsen does not, and cannot, meet because Larsen always uses part of an instruction's address to specify instruction type; thus, the instruction address and instruction type can never be independent of one another. For the

Board's convenience, in section IV, Applicant has provided a chart identifying specific language in each independent claim that distinguishes Applicant's invention from Larsen.

A. Larsen Does Not Disclose The Apparatus Of Claims 1-14

Claims 1-14 each require that a memory access controller, which is used to access program instruction words stored in memory, "not be[] responsive to [the] one or more indicator bits of [a] program counter register" that specify the current instruction set. In Larsen, the three highest-order bits from IAR 3 are used to specify the instruction set (e.g., the bits stored in IDSR 10). However, these bits also make up a portion of the sequence of bits used to specify the address of an instruction in I-store 2. Therefore, any part of the Larsen system responsible for controlling access to I-store 2 would necessarily have to be responsive to the bits used to specify the instruction set, i.e., the three highest-order bits of IAR 3. Such a system clearly does not disclose each and every limitation of claims 1-14.

B. Larsen Does Not Disclose The Method Of Claims 15-20 Or Apparatus Of Claims 51-56

Claims 15-20 and 51-56 each require that "the instruction set identified by the instruction set indicator portion of the sequence of bits is identifiable without regard to the address derived from the address portion of the sequence of bits." (emphasis added.) In Larsen, the identification of an instruction set depends directly on the instruction's location in memory - - "[d]ecoding of any specific instruction in the IDR 1 thus depends not only on the contents of the IDR 1 but on the region of the I-store 2 from which the instruction was fetched. This portion of the address is provided by the IDSR 10 output." (Larsen, col. 6, li. 11-16 (emphasis added).) Thus, the system of Larsen cannot identify an instruction set without regard to the address of the instruction as required by claims 15-20 and 51-56. Such a system thus fails to disclose each and every limitation of claims 15-20 and 51-56.

C. Larsen Does Not Disclose The Methods Of Claims 21-26 and 33-38, Apparatus Of Claims 45-50, Or Architecture Of Claims 57-64

Claims 21-26, 33-38, 45-50 and 57-64 each require that "the instruction set indicator portion [has] at least one bit that is not part of the address portion of the sequence of bits." (emphasis added.) In Larsen, the bits used to indicate an instruction set, i.e., the three highest-order bits from IAR 3, are **all** part of the sequence of bits used to specify an instruction address. In other words, there are no bits used to indicate an instruction set that are not also used to specify an address. Thus, Larsen does not disclose each and every limitation of claims 21-26, 33-38, 45-50 and 57-64.

D. Larsen Does Not Disclose The Apparatus of Claims 27-32 Or Architecture Of Claims 39-44

Claims 27-32 and 39-44 each require that one or more control flags that specify an instruction set "are set without regard to the address derived from the address portion of the sequence of bits." (emphasis added.) In Larsen's partitioned memory approach, the instruction type of a given instruction is specified based on that instruction's location in memory. Thus, to the extent Larsen has a control flag that specifies an instruction set, the flag is defined by and depends directly on a portion of the instruction's address in memory (the high order bits of the address). The instruction sets of Larsen, therefore, are not specified "without regard to the address." Thus, Larsen does not disclose each and every limitation of claims 27-32 and 39-44.

E. Larsen Does Not Disclose The Program Counter Register Of Claim 65

Claim 65 requires that the at least one bit that identifies the instruction set to be used is "not a member of the subset" of bits that specify the instruction address. First, in Larsen, there can be no such subset of bits that specify the instruction address because the address is the entire sequence of bits from IAR 3. Second, in Larsen, because all of the bits of IAR 3 must be used to

specify the address of an instruction, the three highest-order bits, which are re-used to specify an instruction set, are within any sequence of bits that specifies the instruction address. There is no bit in Larsen used to identify the instruction set that is not a member of the bits used to specify the instruction address. Thus, Larsen does not disclose each and every limitation of claim 65.

F. Larsen Does Not Disclose The Method Of Claim 66 Or Apparatus Of Claims 67-70

Claims 66-70 each require setting the value of a flag for identifying an instruction set in response to an instruction where "the value of the flag is not dependent upon the address of the next instruction." (emphasis added). For the same reasons as set forth above with respect to claims 27-32 and 39-44, Larsen cannot disclose this limitation because, to the extent Larsen has a flag that specifies an instruction set, the flag is defined by and depends directly on a portion of the instruction's address in memory (the high order bits of the address). Therefore, Larsen does not disclose each and every limitation of claims 66-70.

IV. Chart Of Independent Claims

For the Board's convenience, Applicant has provided the following chart identifying specific language in each independent claim that distinguishes Applicant's invention from Larsen.[2]

Claim	Claim Language	Larsen
1	"a memory access controller operable to access program instruction words stored in said data memory, said access controller **not being responsive to said one or more indicator bits of said program counter register**"	I-store **2** access controller is responsive to bits used for indicating instruction type – the three highest-order bits from IAR **3** are used to indicate **both** instruction type and a portion of an instruction's address.

[2] In any given claim, other claim limitations may also distinguish Larsen, but the examples provided in the above chart demonstrate that Larsen cannot anticipate Applicant's invention.

Claim	Claim Language	Larsen
15	"accessing a sequence of bits, the sequence of bits having an address portion and an instruction set indicator portion . . . wherein the instruction set identified by the instruction set indicator portion of the sequence of bits is identifiable **without regard to the address derived from the address portion of the sequence of bits**"	Because the "[d]ecoding of any specific instruction . . . depends . . . on the region of the I-store 2 from which the instruction was fetched" (col. 6, li. 11-14), the identification of an instruction set depends directly on the instruction's address. The sequence of bits used to derive an address is also used, at least in part, to identify an instruction set.
21	"accessing a sequence of bits, the sequence of bits having an address portion and an instruction set indicator portion, **the instruction set indicator portion having at least one bit that is not part of the address portion** of the sequence of bits"	As explained above, **all** the bits from IAR 3 used to indicate an instruction set are part of the address portion of the sequence of bits, because the entire sequence of bits in IAR 3 is used to specify an instruction address.
27	"[a] sequence of bits having an address portion and an instruction set indicator portion, [a] processor core deriving an address of a second instruction from the address portion of the sequence of bits and using the instruction set indicator portion of the sequence of bits to set one or more control flags . . . the state of the one ore more control flags specifying a current instruction set . . . **wherein the one or more control flags are set without regard to the address derived from the address portion of the sequence of bits**"	At least a portion of the sequence of bits in IAR 3 used to identify an instruction address are used to set the value of the bits stored in IDSR 10. The bits stored in IDSR 10, in turn, specify the instruction set. Thus, the bits used to specify the instruction set are set **with** regard to the sequence of bits used to derive the instruction address.
33	"a processor core responsive to a first instruction to access a sequence of bits to derive an address of a second instruction, the sequence of bits having an address portion and an instruction set indicator portion and **the instruction set indicator portion having at least one bit that is not part of the address portion** of the sequence of bits, the address of the second instruction being derived from the address portion of the sequence of bits"	See claim 21.

Claim	Claim Language	Larsen
39	"[a] sequence of bits having an address portion and an instruction set indicator portion, [a] processor core deriving an address of a second instruction from the address portion of the sequence of bits and using the instruction set indicator portion of the sequence of bits to set one or more control flags . . . the state of the one ore more control flags specifying a current instruction set . . . **wherein the one or more control flags are set without regard to the address derived from the address portion of the sequence of bits**"	See claim 27.
45	"a processor core responsive to a first instruction to access a sequence of bits to derive an address of a second instruction, the sequence of bits having an address portion and an instruction set indicator portion and **the instruction set indicator portion having at least one bit that is not part of the address portion** of the sequence of bits, the address of the second instruction being derived from the address portion of the sequence of bits"	See claim 21.
51	"means for accessing a sequence of bits in response to a first instruction, the sequence of bits having an address portion and an instruction set indicator portion . . . wherein the instruction set identified by the instruction set portion of the sequence of bits is identifiable **without regard to the address derived from the address portion of the sequence of bits**"	See claim 15.
57	"accessing a sequence of bits comprising an address portion and an instruction set indicator portion in response to the first set of one or more control signals, **the instruction set indicator portion having at least one bit that is not part of the address portion** of the sequence of bits"	See claim 21.

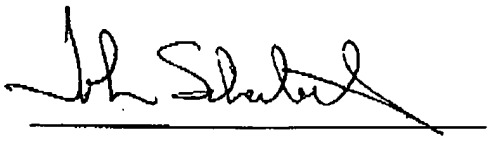
Claim	Claim Language	Larsen
65	"an ordered set of bits; wherein a subset of the ordered set of bits identifies an address of an instruction; and at least one bit of the ordered set of bits identifies an instruction set; and **wherein the at least one bit is not a member of the subset**"	The bits from IAR **3** used to identify an instruction set are a member of the set of bits used to identify an instruction address because **all** bits from IAR **3** are used to identify an instruction address.
66	"inserting the address of a next instruction into a register and setting the value of a flag, where **the value of the flag is not dependent upon the address of the next instruction** [and] selecting an instruction set based upon the value of the flag"	An instruction set is selected based upon the value of the bits in IDSR **10**. The bits in IDSR **10** correspond with at least a portion of the address bits in IAR **3** (e.g., the three highest-order bits of the address), thus the value of the bits in IDSR **10** is dependent upon the address of the next instruction.
67	"a flag for identifying [a] first instruction set; wherein . . . **the value of the flag is not dependent upon the address of the next instruction**"	See claim 66.

V. Conclusion

Applicant respectfully submits that, for the foregoing reasons, Claims 1-70 were improperly rejected as anticipated and are therefore allowable over the cited art. As explained above, Larsen **must** use at least a portion of the address where an instruction resides to determine the instruction type, whereas in every claim of Applicant's invention, the instruction type is determined independent of the address where the instruction resides. Applicant accordingly requests reversal of the rejections.

Dated: July 11, 2008

Respectfully submitted,



John Scheibeler
Reg. No. 35,346
Customer No. 007470
Direct Dial: (212) 819-8830